REMARKS

The claims are claims 1 to 3, 5 to 9, 11 and 17 to 19.

The application has been further amended at many locations to correct minor errors and to present uniform language throughout. The amendments include correction of those errors noted by the Examiner.

The ABSTRACT is amended to conform to the current word limit.

Claims 1 to 3, 5 to 9 and 11 have been amended. Claims 4, 10 and 12 to 14 are canceled. New claims 17 to 19 are added. Claims 1 and 11 are amended in response to the objection due to informalities. Claims 2, 3 and 5 to 9 have been similarly amended. Claims 1 and 11 are further amended to distinguish over the references. Claim 2 has been amended to distinguish over the references. New claims 17 to 19 are added. New claims 17 and 19 recite an output multiplexer taught in this application and illustrates at 148 in Figures 10 to 12. New claim 18 recites a pipeline register disclosed in this application at page 38, lines 10 to 14 and illustrated in Figure 22b.

Claims 1, 8 and 11 as amended are no longer informal. The prior recitations of "first operand input," "second operand input," "first output" and "second output" have been changed to "operand input" and "output." This does not create any ambiguity because the claims already recited "first functional unit group" or "second functional unit group" when later referring to these elements. Claims 1 and 11 are further amended to make clear that "having a first input..." refers to the respective multiplexers and not the comparators. Corresponding amendments are made to claims 2, 3 and 5 to 9. All instances in claims 8 and dependent claim 9 of "third comparator" have been changed to "second comparator." Similarly, all instances of "third multiplexer" have been changed to "second multiplexer."

Claim 1 and 5 to 9 were rejected under 35 U.S.C. 103(a) as made obvious by the combination of O'Connor U.S. Patent No. 6,266,766 and the Hennessy publication. The OFFICE ACTION states t page 7, lines 1 to 13:

> "c. Hennessy has taught on page 188 the use of functional unit groups each including a plurality of functional units. Functional unit group 1, the integer unit, contains functional units for load, store, ALU, and branch operations. Functional unit group 3, the FP adder, contains functional units for floating point add, subtract, and conversion. The other two functional unit groups contain a functional unit for each of floating point and integer multiply or divide operations.
> "d. The multiple functional unit groups allow for more instructions to be executed in parallel because each functional unit group contains multiple functional units. By executing more instructions in parallel, a higher pipeline throughput is achieved and thus making the processor faster.
> "e. The ability to speed up operation of the processor would have motivated one of ordinary skill in the art to use multiple functional unit groups in the design of O'Connor."

The Applicants respectfully submit that this reasoning is incorrect and that the combination of O'Connor and Hennessy do not make these claims obvious.

Claims 1 and 11 recite subject matter not made obvious by the combination of O'Connor and Hennessy. Claims 1 and 11 recite "operate on said received data employing an instruction-specified one of said first functional units" and "operate on said received data employing an instruction-specified one of said second functional units." This recitation of claims 1 and 11 requires that only one of plural first functional units can be used at a time. This also requires that only one of plural second functional units can be used at a time. The Examiner's argument that one skilled in the art would be motivated to provide multiple functional units groups is contrary to this recitation of claims 1 and 11. Providing plural functional units within groups that

cannot execute together would not provide the possibility of executing more instructions in parallel as argued by the Examiner. Thus one skilled in the art would not be motivated as suggested by the Examiner. Accordingly, claims 1 and 11 are allowable over the combination of O'Connor and Hennessy.

Claims 1 and 11 recite further subject matter not made obvious by the combination of O'Connor and Hennessy. Claims 1 and 11 recite "said first functional units of said first functional unit group and said second functional units of said second functional unit group selected whereby functions often executed simultaneously within the same instruction cycle have corresponding functional units placed in different functional unit groups and functions which are not often executed together within the same instruction cycle have corresponding functional units placed in the same functional unit group." This limitation is taught in the application at page 33, line 28 to page 34, line 4. The Applicants respectfully submit that there is no teaching in O'Connor or Hennessy making obvious this limitation. The teaching of this application at page 33, lines 5 to 22 are instructive in this regard. The combination of O'Connor and Hennessy would make obvious providing hot paths for plural functional units. As pointed out in this application, providing hot paths for every functional unit for every group "would require too much hardware and too much time for processing" (application at page 33, lines 14 and 15). The solution of this invention is to provide fewer hot paths. The above quoted limitation of claims 1 and 11 ensures that the functional units are distributed among the functional unit groups so that the provided hot paths have the greatest usefulness. The Applicants respectfully submit that the combination of O'Connor and Hennessy fail to teach this subject matter. Accordingly, claims 1 and 11 are allowable over O'Connor and Hennessy.

Claims 5 to 9 are allowable by dependent upon allowable claim 1.

Claim 2 was rejected under 35 U.S.C. 103(a) as made obvious by the combination of O'Connor U.S. Patent No. 6,266,766, the Hennessy publication and McCullough et al U.S. Patent No. 5,826,069. The OFFICE ACTION cites Figure 3B and column 14, lines 38 to 59 as making obvious the recitation of pipeline stages.

Claim 2 recites subject matter not made obvious by the combination of O'Connor U.S. Patent, Hennessy and McCullough et al. Claim 2 recites "a first pipeline stage consisting of a register read operation from said register file to provide operands for a selected functional unit of said first and second functional unit groups." McCullough et al states at column 14, lines 40 to 45:

> "During the first 1/2 cycle period 121 the data is read by the present invention write logic 45 from the ROB 71 and the data for the set of retiring micro-ops is placed into the queue memory 73; this occurs in 3.3 nanoseconds."

This read from reorder buffer 71 is not the recall of operands for the functional unit groups as recited in claim 2. McCullough et al states at column 10, lines 13 to 25:

> "As shown by FIG. 3(A), a Reorder Buffer 71 receives the results of executed micro-ops over bus 95 and orders the micro-op results according to their original program code order as present within the program code read by the superscalar microprocessor. This ordering process is not pertinent to the present invention and is therefore not described in detail herein. However, it is important to realize that the ROB 71 supplies the present invention, for each clock cycle, with as many as four results (i.e., associated with four separate retiring micro-ops) in program code order. Every clock cycle, the ROB 71 supplies as many as four entries into a temporary storage queue 73 of the present invention logic 45."

This disclosure of McCullough et al teaches that the data recalled during the first one half cycle from ROB 71 is not the operands for the functional unit groups but "the micro-op results." In addition, ROB 71 is not the register file as claimed but a buffer which reorders results into program code order. Note particularly that Figure 3A of McCullough et al illustrates real register file and flag words 83 as a separate structure than ROB 71. Accordingly, claim 2 is allowable over the combination of O'Connor U.S. Patent, Hennessy and McCullough et al.

Claim 2 recites further subject matter not made obvious by the combination of O'Connor U.S. Patent, Hennessy and McCullough et al. Claim 2 recites "a first pipeline stage consisting of...a first half of operation of said selected functional unit of said first and said second functional unit groups, and a second pipeline stage consisting of a second half of operation of said selected functional unit of said first and said second functional unit groups." The guarantee bits produced by event logic 85 are not the recited first and second half of the functional unit operation. McCullough et al states at column 13, lines 35 to 59:

> "Also input to the priority stage 79 of FIG. 3(A) of the present invention are four guarantee bits over bus 90 that originate from an event logic block 85. There are four possible guarantee bits, one for each of the possible retiring micro-ops of the queue 73. The event logic block 85 receives inputs 93 from the microprocessor and indicate when certain errors or conditions take place that may bar the write update operation (and therefore bar the register file update) of certain of the retiring micro-ops of the queue 73. For instance, if uop2 generated a divide by zero error, then the result of uop3 should not be written to the register file and the guarantee bit for uop3 would not be set. In effect, each guarantee bit for an associated micro-op signifies that the associated uop is 'guaranteed' for proper execution and guaranteed for a write operation into the register file 83. If a micro-op is to be denied a write operation because of the event logic 85, it is likely to be a later-in-time micro-op (i.e., a high priority micro-op). The four guarantee bits are

> fed into the priority logic because if the higher priority micro-ops should not update, then other lower priority micro-ops should be able to update in lieu of the denied micro-op (i.e., take the priority of the denied micro-op). Therefore, if the guarantee bit for a particular micro-op is not set, the present invention write logic 45 will ignore the write operation for that micro-op."

This disclosure on McCullough et al makes clear that event logic block 85 does not provide the results of the functional unit. Instead event logic block 85 indicates whether the results stored in ROB 71 corresponding to a particular retiring micro-op are valid. Note that the above quoted portion of column 10 of McCullough et al states that the results have already been computed and are stored in ROB 71. Accordingly, this disclosure of McCullough et al fails to make obvious the above quoted recitations of claim 2.

Claim 3 recites subject matter not made obvious by the combination of O'Connor U.S. Patent, Hennessy and Olson et al. Claim 3 recites "said first comparator further receives an indication of said second destination register number of a second preceding instruction, said first comparator further indicating whether said first operand register number of said current instruction matches said second destination register number of said second preceding instruction." By this limitation the first comparator determines a match not detected by O'Connor or Olson et al. Olson et al states at column 4, lines 8 to 22:

> "The address comparators 24 and 26 compare these REG FILE 16 addresses to determine if the REG FILE 16 register selected for access during a current cycle is equal to the REG FILE 16 register selected for update during the cycle. If these two addresses are found to be equal it is indicated that the result of an ALU operation during an instruction cycle N is to be used as an operand for an ALU operation during the subsequent cycle N+1. When this condition is detected the output of the associated address comparator 24 and 26 enables

> the corresponding multiplexer MUX 18 or MUX 20 select (S) input to gate the ALU 22 result on CB00:31 directly to the corresponding input of the ALU 22, thereby effectively bypassing the REG FILE 16."

The Applicants respectfully submit that the "register selected for update" corresponds to the destination register of the preceding instruction. Thus Olson et al fails to make a comparison recited in claim 3. Accordingly, claim 3 is allowable over the combination of O'Connor U.S. Patent, Hennessy and Olson et al.

Claims 17 to 19 recite subject matter not anticipated by nor made obvious by the cited references. Claims 17 and 19 each recite an output multiplexer "selecting said output of said instruction-selected one of said first functional units" in the first functional unit group and a similar output multiplexer as part of the second functional unit group. The Applicants submit that none of the cited references disclose or make obvious this subject matter. Claim 18 recites "a pipeline latch in the middle for latching a logical state of said functional unit between said first pipeline stage and said second pipeline stage." This subject matter is likewise neither anticipated or made obvious by any cited reference.

The Applicants respectfully submit that all the present claims are allowable for the reasons set forth above. Therefore early reconsideration and advance to issue are respectfully requested.

If the Examiner has any questions or other correspondence regarding this application, Applicants request that the Examiner contact Applicants' attorney at the below listed telephone number and address to facilitate prosecution.

Texas Instruments Incorporated
P.O. Box 655474 M/S 3999
Dallas, Texas 75265
(972) 917-5290
Fax: (972) 917-4418

Respectfully submitted,

Robert D. Marshall, Jr.
Reg. No. 28,527